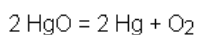


Enunciado

La reacción correspondiente a la descomposición del **óxido mercurico** es:



Hallar la **cantidad** de oxígeno (en gramos) que puede obtenerse al calentar 37 g de HgO

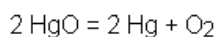
Nota: la precisión del valor pedido es ± 0.01

Respuesta

Introduce la respuesta:

Enunciado

La reacción correspondiente a la descomposición del **óxido mercurico** es:



Hallar la **cantidad** de oxígeno (en gramos) que puede obtenerse al calentar 70 g de HgO

Nota: la precisión del valor pedido es ± 0.01

Respuesta

Introduce la respuesta:

AulaWeb

Creación de preguntas de enunciado variable

Ángel García-Beltrán y Raquel Martínez

Universidad Politécnica de Madrid

Julio de 2008

AulaWeb – Creación de preguntas de enunciado variable
A. García-Beltrán, R. Martínez Fernández
I.S.B.N.: 978-84-691-4776-4
(c) Madrid, 2008
e-mail: {agarcia, raquelm}@etsii.upm.es
Tel.: 91 336 4266, 3200
URL: <http://www.dii.etsii.upm.es/aulaweb>

Índice

1.	Introducción	4
2.	Esquema del programa enunciado/corrector.....	5
3.	Modos de funcionamiento	7
4.	Un ejemplo de programa para una pregunta de enunciado variable.....	9
5.	Introducción de una pregunta de enunciado variable en AulaWeb	13
6.	Ejemplos de preguntas de enunciado variable	17
7.	Pregunta de selección simple (ejemplo 1)	18
8.	Pregunta de selección simple (ejemplo 2)	22
9.	Pregunta de selección múltiple (ejemplo 3).....	26
10.	Pregunta de respuesta numérica entera (ejemplo 4)	30
11.	Pregunta de respuesta numérica real (ejemplo 5)	33
12.	Conclusión	36
	Bibliografía	37

Sobre este manual

La presente edición de este Manual de Creación de Preguntas de Enunciado Variable hace referencia al sistema AulaWeb implantado en la ETSI Industriales en abril de 2007 y trata de algunos elementos de la plataforma AulaWeb no recogidos en los anteriores Manuales del Profesor. Obviamente tampoco recoge las funcionalidades del sistema añadidas con posterioridad.

Participantes en el proyecto

Las siguientes personas han colaborado en el proyecto de desarrollo de las distintas versiones del sistema: Aurora Alonso, José María Arranz, Pablo Avendaño, Marcos Aza, Juan Antonio Criado, Francisco de Ory, Carlos Engels, Miguel Fernández, Vanesa Gámiz, Pilar García, Ángel García-Beltrán, Manuel González, Jorge Granado, Teresa Hernández, Isadora Iglesias, José Alberto Jaén, Antonio R. López, Diego López, Julio A. Martín, Marcos Martín, Raquel Martínez, Francisco José Mascató, David Molina, Carlos Moreno, Daniel J. Muñoz, Luis Miguel Pabón, Juan Carlos Pérez, Alberto Rodelgo, Santiago Tapia, Alberto Valero, Eduardo Villalar y Carlos Zoido dentro de la División de Informática Industrial de la ETSII-UPM.

1. Introducción

El sistema de autoevaluación de AulaWeb se apoya en una base de datos de preguntas que el usuario profesor puede generar antes de configurar los ejercicios. Las preguntas pueden ser de distintos tipos: tipo test de respuesta única entre varias opciones (selección simple), tipo test de respuesta múltiple entre varias opciones (selección múltiple), de respuesta numérica entera, de respuesta numérica real, de respuesta de cadena de caracteres, de código de programa, etcétera. Las preguntas variantes o de enunciado variable son cuestiones en las que se puede modificar de forma dinámica algunos datos del enunciado, de forma que a partir de un problema tipo puedan formularse un número indefinido de versiones. Por ejemplo, el siguiente enunciado podría ser el de una pregunta con parámetros o datos variables:

“Se construye un árbol binario que se mantiene permanentemente ordenado y equilibrado en el sentido AVL, mediante la inserción sucesiva de los elementos: 7, 5, 15, 13, 11, 9 y 4. ¿Cuál es el elemento que queda como nodo raíz?”

donde los elementos subrayados serían los datos variables, de tal manera que cada vez que la pregunta se selecciona de la base de datos para formar parte de un ejercicio, los valores de dichos datos son diferentes. En la Figura 1 se muestra un ejemplo de dos versiones de enunciados distintos generados a partir de la misma pregunta.

Enunciado
Se construye un **árbol binario que se mantiene permanentemente ordenado y equilibrado en sentido avl** formado por la inserción sucesiva de elementos con claves 18, 14, 30, 29, 1, 10 y 2. Indicar cuál es el elemento que queda como **nodo raíz**.

Respuesta
Introduce la respuesta:

Enunciado
Se construye un **árbol binario que se mantiene permanentemente ordenado y equilibrado en sentido avl** formado por la inserción sucesiva de elementos con claves 40, 4, 7, 9, 37, 10 y 28. Indicar cuál es el elemento que queda como **nodo raíz**.

Respuesta
Introduce la respuesta:

Figura 1. Dos versiones distintas de la misma pregunta de enunciado variable

En el módulo de autoevaluación de AulaWeb y para cada pregunta de este nuevo tipo (tipo VA o *Variante*), se realiza un pequeño programa que se encarga, por un lado, de generar el texto del enunciado para ser incorporado a la página ASP, y por otro, de corregir la respuesta del alumno cuando éste responde a la pregunta. Gracias a la modularidad de la base de datos y al sistema de autoevaluación, la incorporación de este nuevo tipo de preguntas es muy sencilla. Lo que es algo más complicado es realizar el propio programa enunciador/corrector. Para simplificar esa tarea se propone una plantilla de programa escrito en TurboPascal en la siguiente sección.

2. Esquema del programa enunciado/corrector

A continuación se muestra lo que sería el esqueleto de un programa enunciador/corrector para preguntas de este tipo. Se ha elegido el lenguaje de programación TurboPascal 7.0 por la facilidad para su aprendizaje y legibilidad aunque podría haberse escrito en cualquier otro lenguaje de programación. Las partes en cursiva del código fuente del programa serían las únicas que habría que reescribir para cada pregunta, ya que el resto es común.

```

Program Variante;

Var
  i,error:integer;
  semilla:longint;
  programa,accion,respuesta,enunciado,entrada:string;
  { Datos específicos . . . }

Procedure GeneraDatos;
begin
  RandSeed:=semilla;
  { Generación los datos aleatorios del enunciado . . . }
end;

Procedure GeneraEnunciado;
begin
  writeln('bla bla bla');
end;

Procedure GeneraInput;
begin
  writeln('Introduce la respuesta: <input id="t1" type="text" name="respuesta" size="16"
maxlength="32">');
end;

Procedure GeneraHTML;
begin
  writeln('<center><table border=0 cellpadding=0 cellspacing=8 width="100%"><tr>');
  writeln('<td id="t1" valign="top">');
  writeln('<b>Enunciado</b><br>');
  writeln('<div id="t1" style="text-align: justify;">');
  GeneraEnunciado;
  writeln('</div><br></td></tr><tr><td id="t1" valign="top">');
  writeln('<b>Respuesta</b><br>');
  GeneraInput;
  writeln('</td></tr></table></center>');
end;

Procedure Correcta(flag:boolean);
begin
  if flag then writeln('OK. Respuesta correcta')
  else writeln('ERROR. Respuesta incorrecta');
end;

Procedure CorrigeRespuesta;
var s:string;
begin
  { Comparaciones entre la respuesta dada y la solucion y operaciones de corrección }
end;

BEGIN

  { Obtiene los parametros de entrada }
  programa:=ParamStr(0);

```

```

accion:=ParamStr(1);
val(ParamStr(2),semilla,error);
if (accion='/c') or (accion='/C') then respuesta:=ParamStr(3);

{ Genera los datos aleatorios }
GeneraDatos;

{ En caso de acción '/e', genera el enunciado }
if (accion='/e') or (accion='/E') then GeneraHTML;

{ En caso de acción '/c', corrige la respuesta }
if (accion='/c') or (accion='/C') then CorrigeRespuesta;

```

END.

Básicamente, habrá que implementar el código que lleve a cabo las siguientes operaciones:

- Declarar las variables particulares de la pregunta. Por ejemplo, si es necesaria una lista de diez números enteros, habrá que declarar algo como `datos:array[1..10] of integer`;
- Iniciar los datos aleatorios en la rutina `GeneraDatos`, que ya establece la semilla adecuada. Por ejemplo: `for i:=1 to 10 do datos[i]:=random(100)`;
- Generar el texto completo del enunciado, incluyendo las representaciones pertinentes de los datos, en la función `GeneraEnunciado`.
- Generar el campo de petición de datos en la función **GeneraInput**. Algunos ejemplos típicos del texto a generar son los siguientes:

Si se pide un número:

```
'Introduce la respuesta: <input id=t1 type=text name=respuesta size=16
maxlength=32>';
```

Si se pide una cadena o lista de elementos:

```
'Introduce la respuesta: <input id=t1 type=text name=respuesta size=32
maxlength=64>';
```

Si se pide una selección simple

```
'Elige la respuesta correcta:<br>'
'<input id=t1 type=radio name=respuesta value=1>opción 1<br>';
'<input id=t1 type=radio name=respuesta value=2>opción 2<br>';
etc.
```

Si se pide una selección múltiple

```
'Elige las respuestas correctas:<br>';
'<input id=t1 type=checkbox name=respuesta value=1>opción 1<br>';
'<input id=t1 type=checkbox name=respuesta value=2>opción 2<br>';
etc.
```

- Implementar el código necesario para corregir la pregunta en función de los datos aleatorios y de la entrada del usuario (variable `respuesta`).

* Nota: la rutina **GeneraHTML** se encarga de combinar en enunciado, el campo de entrada y de generar el fragmento de código HTML que necesitan las páginas ASP que leen y muestran la pregunta. Está en el mismo formato que el resto de los tipos de preguntas estándar.

3. Modos de funcionamiento

La llamada al programa de cada pregunta se llevará a cabo con un parámetro que permite tres modos diferentes de funcionamiento:

a) Generación del enunciado (opción /E)

Al programa se le pasan como parámetros el comando **/E** y un **valor numérico entero** como *semilla* para el generador de números aleatorios. El programa debe devolver, por la salida estándar, el enunciado particular para el caso determinado por la semilla y los campos necesarios para la respuesta en formato HTML (con una estructura de maquetación determinada).

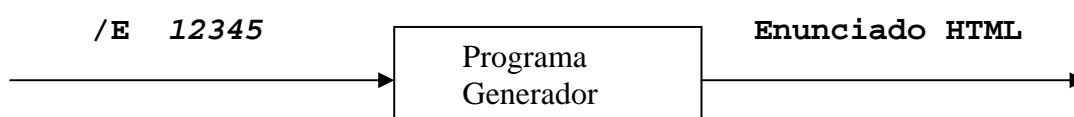


Figura 2. Esquema de funcionamiento del programa para generar el enunciado de la pregunta

Si se utiliza la plantilla propuesta en la sección anterior:

- El cuerpo principal del programa se encarga de recoger los parámetros de entrada.
- En la función `GeneraDatos`, se aplica la semilla aleatoria. Se deben incluir las inicializaciones de todos los parámetros y variables necesarios.
- En la función `GeneraEnunciado`, se debe incluir el texto del enunciado (sin los campos de respuesta).
- En la función `GeneraInput`, se crean el campo (o los campos) de formulario HTML para la respuesta del alumno.
- La función `GeneraHTML` (que no hay que modificar), es la encargada de combinar el enunciado y los campos de respuesta en el formato HTML requerido.

b) Obtención de la solución (opción /S)

Al programa se le pasan como parámetros el comando **/S** y un **valor numérico entero** como *semilla* para el generador de números aleatorios. El programa debe devolver, por la salida estándar, un texto que represente la solución de la pregunta para el caso particular determinado por la semilla. Esta opción no requiere apenas implementación adicional y se utiliza para la comprobación de preguntas en la base de datos.

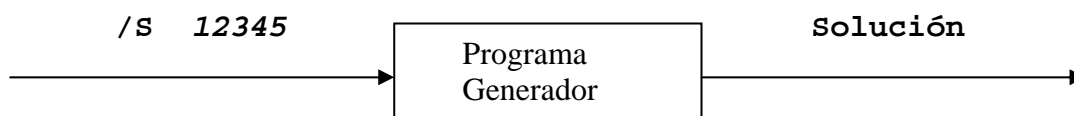


Figura 3. Esquema de funcionamiento del programa para obtener la solución de la pregunta

Si se utiliza la plantilla propuesta en la sección anterior:

- La función `EnviaSolucion` es la encargada de escribir la solución (generalmente basta con escribir el contenido de la variable solución o alguna similar).

c) Corrección de la respuesta del alumno (opción `/C`)

Al programa se le pasan como parámetros el comando `/C`, un **valor numérico entero** como semilla para el generador de números aleatorios, y uno o más parámetros que constituyen la respuesta que ha dado el alumno. El programa debe devolver, por la salida estándar, la corrección al caso particular de la pregunta, para la semilla y la respuesta dadas. Dicha corrección consiste en escribir las cadenas “OK” o “ERROR” según el caso.

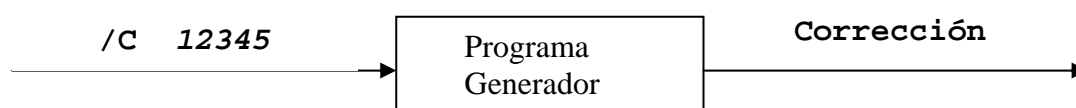


Figura 4. Esquema de funcionamiento del programa para corregir la respuesta dada por el alumno

Si se utiliza la plantilla propuesta en la sección anterior:

- En la función `CorrigeRespuesta`, se deben realizar las comparaciones pertinentes entre la respuesta emitida por el alumno y la solución real.
- La función `Correcta`, se encarga de escribir el mensaje adecuado según la respuesta sea correcta o no.

4. Un ejemplo de programa para una pregunta de enunciado variable

En esta sección se muestra un ejemplo muy sencillo de una programa para una pregunta de enunciado variable. El enunciado de la pregunta a generar sería la siguiente:

Enunciado

Indicar la salida por pantalla del siguiente programa:

```
program datos;
type archivov1 = file of shortint;
      archivov2 = file of byte;
var f : archivov1;
    g : archivov2;
    i : shortint;
    j : byte;
begin
  i:=5;
  assign(f,'data');
  rewrite(f);
  for i:= -3 to 2 do write(f,i);
  close(f);
  assign(g,'data');
  reset(g);
  seek(g,i);
  read(g,j);
  writeln(j);
  close(g);
end.
```

Respuesta

donde los valores numéricos enteros correspondiente al valor inicial y al valor final del bucle for (-3 y 2) pueden variar.

El programa quedaría como sigue (de nuevo se muestran en cursiva las únicas partes que habría que realizar):

Program

Variante;

Var

```
i,error:integer;
semilla:longint;
programa,accion,respuesta,enunciado,entrada:string;
{ Datos especificos }
a,b:integer;
```

```

Procedure GeneraDatos;
begin
  RandSeed:=semilla;
  { generacion los datos aleatorios del enunciado }
  a:=-1-random(20);
  b:=1+random(20);
end;

Procedure GeneraEnunciado;
begin
  writeln('Indicar la salida por pantalla del siguiente programa:');
  writeln('<pre>program datos;');
  writeln('type archivo1 = file of shortint;');
  writeln('      archivo2 = file of byte;');
  writeln('var f : archivo1;');
  writeln('      g : archivo2;');
  writeln('      i : shortint;');
  writeln('      j : byte;');
  writeln('begin');
  writeln('i:=5;');
  writeln('assign(f,'data');');
  writeln('rewrite(f);');
  writeln('for i:= ',a,' to ',b,' do write(f,i);');
  writeln('close(f);');
  writeln('assign(g,'data');');
  writeln('reset(g);');
  writeln('seek(g,i);');
  writeln('read(g,j);');
  writeln('writeln(j);');
  writeln('close(g);');
  writeln('end.</pre>');
end;

Procedure GeneraInput;
begin
  writeln('Introduce la respuesta: <input id="t1" type="text" name="respuesta"
size="16" maxlength="32">');
end;

Procedure GeneraHTML;
begin
  writeln('<center><table border=0 cellpadding=0 cellspacing=8
width="100%"><tr>');
  writeln('  <td id="t1" valign="top">');
  writeln('    <b>Enunciado</b><br>');
  writeln('  <div id="t1" style="text-align: justify;">');
  GeneraEnunciado;
  writeln('  </div><br></td></tr><tr><td id="t1" valign="top">');
  writeln('  <b>Respuesta</b><br>');
  GeneraInput;
  writeln('  </td></tr></table></center>');
end;

Procedure Correcta(flag:boolean);
begin
  if flag then writeln('OK. Respuesta correcta')
  else writeln('ERROR. Respuesta incorrecta');
end;

Procedure CorrigeRespuesta;

```

```
var s:string;
begin
  str(a+b,s);
  if (respuesta=s) then correcta(true)
  else correcta(false);
end;

BEGIN

  { Obtiene los par metros de entrada }
  programa:=ParamStr(0);
  accion:=ParamStr(1);
  val(ParamStr(2),semilla,error);
  if (accion='/c') or (accion='/C') then respuesta:=ParamStr(3);

  { Genera los datos aleatorios }
  GeneraDatos;

  { En caso de acción '/e', genera el enunciado }
  if (accion='/e') or (accion='/E') then GeneraHTML;

  { En caso de acción '/c', corrige la respuesta }
  if (accion='/c') or (accion='/C') then CorrigeRespuesta;

END.
```

Una vez editado y compilado el programa, se obtiene el archivo ejecutable del programa. Un ejemplo de llamada al programa para que genere el enunciado sería:

```
metxxxxx.exe /e 953485
```

que se corregiría (suponiendo que el alumno ha respondido “13”) con la siguiente llamada al programa en la línea de comando:

```
metxxxxx.exe /e 953485 13
```

Llamándolo con otros valores de la semilla, se obtendrían variantes distintas. Como se ve, la parte más laboriosa es la programación del código que corrige la respuesta, ya que no siempre serán tan sencillos y requerirán distintas estrategias. Por ejemplo, para una pregunta de este tipo:

*Indicar cuáles de los siguientes identificadores corresponden a **procedimientos de apertura** de alguno o algunos de los diferentes tipos de variables archivo en el lenguaje de programación TurboPascal 7.0:*

- (a) assign
- (b) reset
- (c) rewrite
- (d) append
- (e) write
- (f) Ninguno de los anteriores

donde las opciones variasen, habría que tener una lista de identificadores (algunos correctos y otros incorrectos) para poder escoger, evitar las repeticiones en las opciones, etc.

En cualquier caso, a partir de la plantilla establecida, sólo hay que implementar el código efectivo de corrección y, aunque se requiera un esfuerzo en esta tarea, se obtienen, virtualmente, infinitas variaciones. En la mayoría de los casos merecerá la pena este esfuerzo invertido en la generalización frente a la introducción manual de casos particulares de un mismo tipo de pregunta. En la siguiente sección se muestra el proceso de introducción de una pregunta de enunciado variable en la base de datos del sistema AulaWeb y en las sucesivas otros ejemplos de programas para pregunta de enunciado variable.

5. Introducción de una pregunta de enunciado variable en AulaWeb

Acceso: Contenidos -> Solapa Preguntas

La ventana para la gestión de preguntas de la base de datos se muestra en la Figura 5.

Sólo Profesores **Contenidos** 1016 - Informática (A)

Recursos **Preguntas** Curso online Búsqueda

Gestión de preguntas

Puede añadir nuevas preguntas e importar preguntas de otras asignaturas.

También puede localizar todas las preguntas de la base de datos que coincidan con el criterio deseado.

Código *	Claves (Separadas por comas)
<input type="text"/>	<input type="text"/>
En el capítulo	Del tipo
(Todos) ▼	(Todos) ▼
Dificultad	Ordenar por
(Cualquiera) ▼ <input checked="" type="radio"/> Relativa <input type="radio"/> Global	Capítulo ▼
<input type="button" value="Buscar preguntas"/> <input type="button" value="Ver Estadísticas de la base de datos"/>	
<input type="button" value="Exportar Preguntas a XML"/>	

(*) Dejar en blanco para no especificar ninguno.

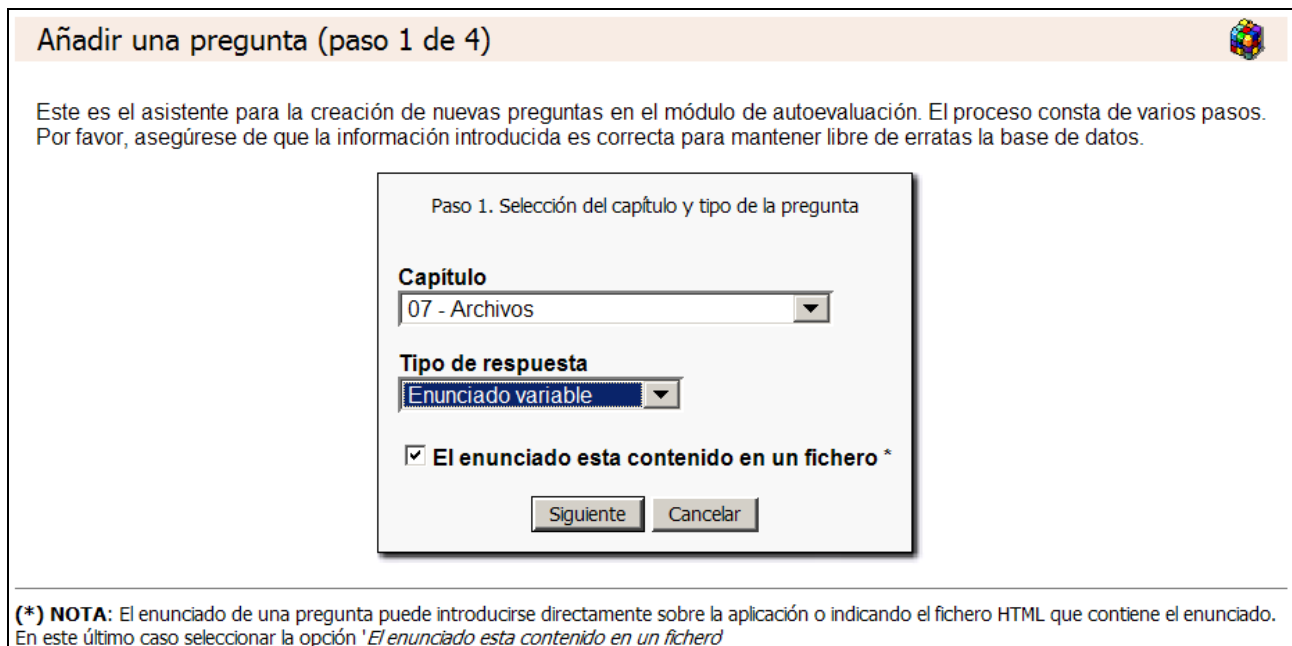
14:53 Curso Académico 2008/09 Profesor: ÁNGEL GARCÍA Y BELTRÁN (Profesor administrador) ID: agarcia

Figura 5. Ventana de gestión de preguntas de la base de datos del entorno del profesor

Desde esta ventana pueden realizarse diferentes tareas de gestión de la base de datos de preguntas, por ejemplo: introducción de nuevas preguntas, importación de preguntas desde otro capítulo o desde otras asignaturas, búsqueda de preguntas existentes, modificación de preguntas ya creadas o visualización de estadísticas de uso de la base de datos de preguntas.

Para introducir una nueva pregunta en la base de datos, se pulsa el botón **Añadir Pregunta** y se siguen los siguientes pasos del asistente:

En el primer paso se selecciona el capítulo del temario y el tipo de pregunta/respuesta. En este caso hay que especificar **Enunciado variable** (Figura 6).



Añadir una pregunta (paso 1 de 4)

Este es el asistente para la creación de nuevas preguntas en el módulo de autoevaluación. El proceso consta de varios pasos. Por favor, asegúrese de que la información introducida es correcta para mantener libre de erratas la base de datos.

Paso 1. Selección del capítulo y tipo de la pregunta

Capítulo
07 - Archivos

Tipo de respuesta
Enunciado variable

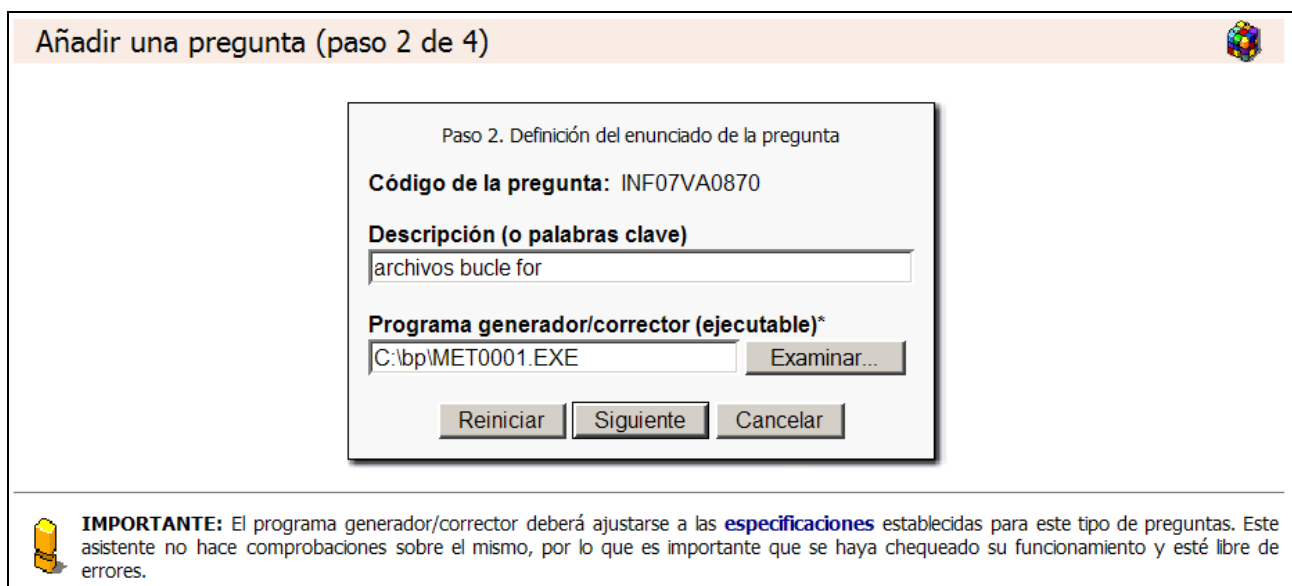
☒ El enunciado esta contenido en un fichero *

Siguiete Cancelar

(*) **NOTA:** El enunciado de una pregunta puede introducirse directamente sobre la aplicación o indicando el fichero HTML que contiene el enunciado. En este último caso seleccionar la opción 'El enunciado esta contenido en un fichero'

Figura 6. Paso 1 para la introducción de una nueva pregunta de enunciado variable

En el segundo paso se facilita una descripción o palabras clave de la pregunta (para facilitar posteriores búsquedas en la base de datos) y el archivo ejecutable que contiene el programa compilado (Figura 7).



Añadir una pregunta (paso 2 de 4)

Paso 2. Definición del enunciado de la pregunta

Código de la pregunta: INF07VA0870

Descripción (o palabras clave)
archivos bucle for

Programa generador/corrector (ejecutable)*
C:\bp\MET0001.EXE Examinar...

Reiniciar Siguiete Cancelar


 **IMPORTANTE:** El programa generador/corrector deberá ajustarse a las **especificaciones** establecidas para este tipo de preguntas. Este asistente no hace comprobaciones sobre el mismo, por lo que es importante que se haya chequeado su funcionamiento y esté libre de errores.

Figura 7. Paso 2 para la introducción de una nueva pregunta de enunciado variable

En el tercer paso se envía el archivo con el código fuente del programa y se asigna el nivel de dificultad de la pregunta (Figura 8).

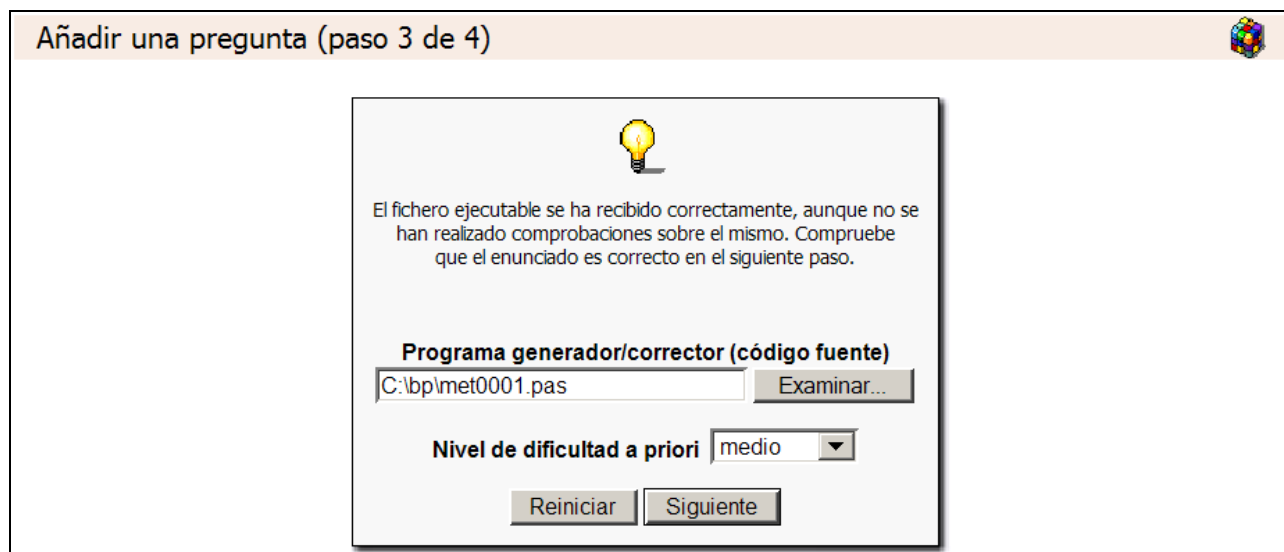


Figura 8. Paso 3 para la introducción de una nueva pregunta de enunciado variable

En la ventana del cuarto y último paso se muestran los datos asociados a la pregunta para su correspondiente confirmación (Figura 9). Al pulsar sobre el enlace del enunciado es posible obtener una primera visualización del enunciado de la pregunta.

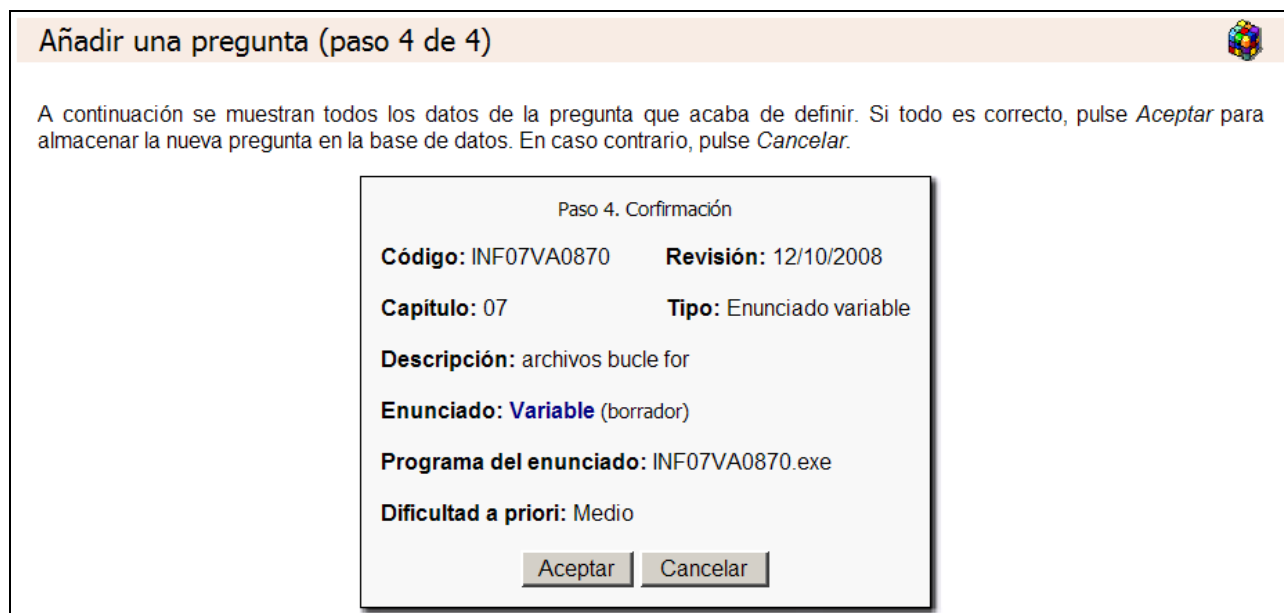


Figura 9. Paso 4 para la introducción de una nueva pregunta de enunciado variable

Al pulsar el botón **Aceptar** se finaliza el proceso de introducción y se muestra una ventana de confirmación del alta de la nueva pregunta en la base de datos (Figura 10).

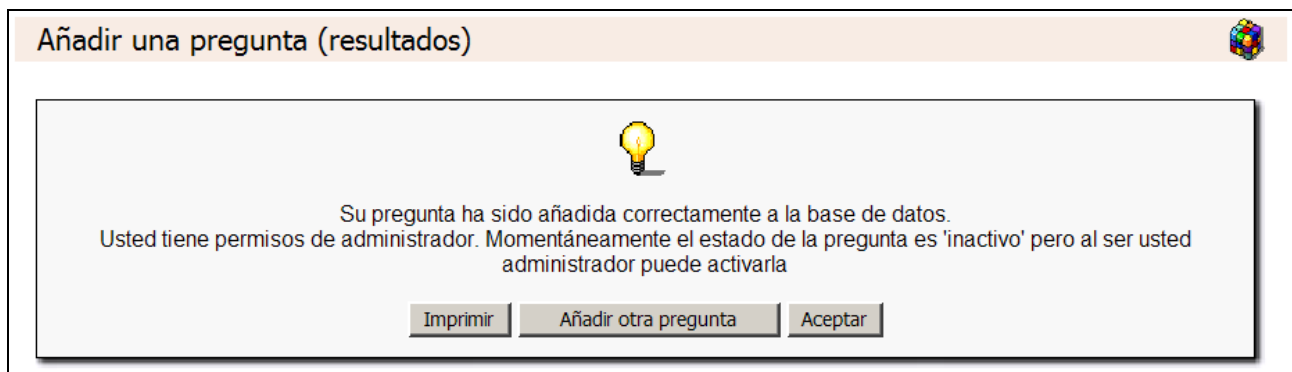


Figura 10. Ventana de confirmación del alta de la pregunta en la base de datos del sistema

La búsqueda y visualización de la nueva pregunta en la base de datos se puede realizar ya desde la ventana de gestión de contenidos. La Figura 11 muestra la ventana correspondiente en el caso de la pregunta recién incorporada.

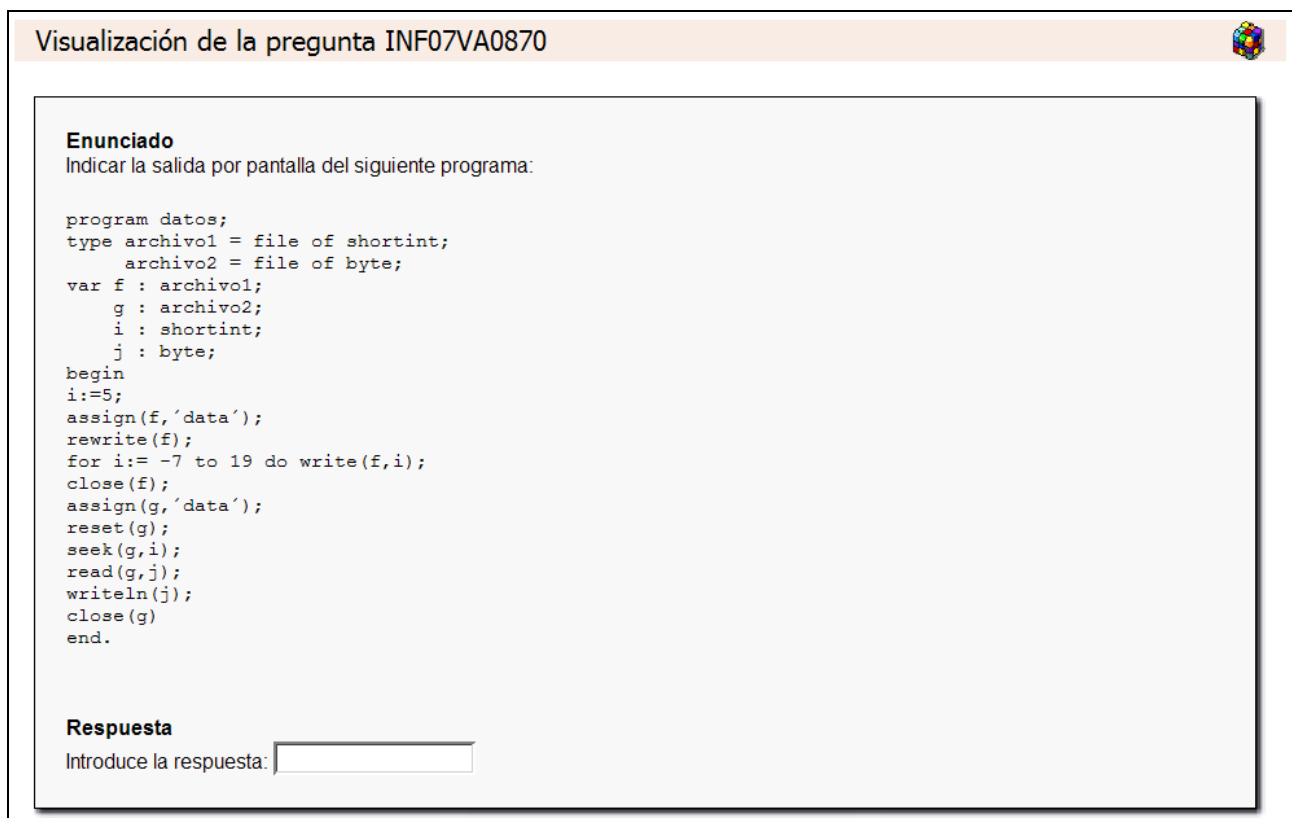


Figura 11. Visualización de una pregunta de enunciado variable

6. Ejemplos de preguntas de enunciado variable

En las siguientes secciones se muestran más ejemplos de preguntas de enunciado variable que permiten diferentes tipos de respuestas. En concreto se exponen preguntas de selección simple, de selección múltiple, de respuesta numérica entera y de respuesta numérica real, aunque el tipo de preguntas que se pueden realizar no tiene casi restricción en la práctica, excepto la habilidad del autor para la creación del enunciado y del programa generador/corrector correspondiente.

Para facilitar la reutilización del código fuente el tipo de letra **negrita** indica el código a modificar del programa para generar otra pregunta similar.

7. Pregunta de selección simple (ejemplo 1)

En este ejemplo se genera una pregunta de **selección simple** en la que tanto una parte del enunciado (una formula química) como las posibles opciones de respuesta son variables. En este caso se juega con siete posible pares de formulas químicas (de hidrocarburos lineales saturados o alcanos) y elementos químicos correspondientes. En la Figura 12 se muestran dos posibles variantes de la pregunta.

Enunciado

La formula quimica $\text{CH}_3\text{-CH}_3$ corresponde al ...

Respuesta

Selecciona la respuesta correcta:

1. ☐ heptano
2. ☐ propano
3. ☐ butano
4. ☐ etano
5. ☐ Ninguna de las anteriores

Enunciado

La formula quimica $\text{CH}_3\text{-CH}_2\text{-CH}_2\text{-CH}_2\text{-CH}_2\text{-CH}_3$ corresponde al ...

Respuesta

Selecciona la respuesta correcta:

1. ☐ metano
2. ☐ hexano
3. ☐ propano
4. ☐ butano
5. ☐ Ninguna de las anteriores

Figura 12. Ejemplo de pregunta de enunciado variable de selección simple

A continuación se muestra el código fuente del correspondiente programa generador/corrector de la pregunta anterior.

```

{ *****
  PREGUNTA:      quimical.pas
  AUTOR:   agarcia@etsii.upm.es
  REVISION: 22.11.2002
  DESCRIPCION: Formulas quimicas y nombres de hidrocarburos
  PARAMETROS VARIANTES: formulas - nombres
  ***** }

Program
  quimical;

Const
  NOpciones=5;      { N$ de opciones que aparecen en el enunciado }
  NDefiniciones=7;   { N$ de pares formulas-nombre }

  secciones:Array[0..NDefiniciones-1] Of String=
    ('CH<sub>4</sub>',
     'CH<sub>3</sub>-CH<sub>3</sub>',
     'CH<sub>3</sub>-CH<sub>2</sub>-CH<sub>3</sub>',
     'CH<sub>3</sub>-CH<sub>2</sub>-CH<sub>2</sub>-CH<sub>3</sub>',
     'CH<sub>3</sub>-CH<sub>2</sub>-CH<sub>2</sub>-CH<sub>2</sub>-CH<sub>3</sub>',
     'CH<sub>3</sub>-CH<sub>2</sub>-CH<sub>2</sub>-CH<sub>2</sub>-CH<sub>2</sub>-
CH<sub>
>3</sub>',
     'CH<sub>3</sub>-CH<sub>2</sub>-CH<sub>2</sub>-CH<sub>2</sub>-CH<sub>2</sub>-
CH<sub>
>2</sub>-CH<sub>3</sub>');
  definiciones:Array[0..NDefiniciones-1] Of String=
    ('metano',
     'etano',
     'propano',
     'butano',
     'pentano',
     'hexano',
     'heptano');

Var
  i,j,error:Integer;
  semilla:LongInt;
  programa,accion,respuesta,enunciado,entrada,solucion:String;
  { Datos especificos }
  sol:Integer;

Procedure GeneraDatos;
Var flag:Boolean;
  txt:String;
Begin
  RandSeed:=semilla;
  { generaci n los datos aleatorios del enunciado }
  sol:=1+Random(NOpciones);
  Str(sol,solucion);
  For i:=0 To NDefiniciones-1 Do Begin
    j:=Random(NDefiniciones-1);
    txt:=secciones[i];
    secciones[i]:=secciones[j];
    secciones[j]:=txt;
    txt:=definiciones[i];
    definiciones[i]:=definiciones[j];
    definiciones[j]:=txt;
  End;
End;

```

```

Procedure GeneraEnunciado;
Begin
  Write('La formula quimica ');
  Write(secciones[sol]);
  Write(' corresponde al ...');
End;

Procedure GeneraInput;
Begin
  WriteLn('Selecciona la respuesta correcta:<br>');
  WriteLn('<table border=0 cellpadding=0 cellspacing=0>');
  For i:=1 To NOpciones-1 Do Begin
    Write('<tr><td id=t1>');
    Write(i, '. <input id=t1 type=radio name=respuesta value=', i, '> ', defini
ciones[i]);
    WriteLn('</td></tr>');
  End;
  Write('<tr><td id=t1>');
  Write(NOpciones, '. <input id=t1 type=radio name=respuesta value=', NOpciones, '>
');
  Write(' Ninguna de las anteriores');
  WriteLn('</td></tr>');
  WriteLn('</table>');
End;

Procedure GeneraHTML;
Begin
  WriteLn('<center><table border=0 cellpadding=0 cellspacing=8
width="100%"><tr>');
  WriteLn('<td id="t1" valign="top">');
  WriteLn('<b>Enunciado</b><br>');
  WriteLn('<div id="t1" style="text-align: justify;">');
  GeneraEnunciado;
  WriteLn('</div><br></td></tr><tr><td id="t1" valign="top">');
  WriteLn('<b>Respuesta</b><br>');
  GeneraInput;
  WriteLn('</td></tr></table></center>');
End;

Procedure Correcta(flag:Boolean);
Begin
  If flag Then WriteLn('OK. Respuesta correcta')
  Else WriteLn('ERROR. Respuesta incorrecta');
End;

Procedure CompactaEspacios(Var cadena:String);
Var i:Integer;
    txt:String;
Begin
  txt:='';
  For i:=1 To Length(cadena) Do
    If (cadena[i]<>' ') Then txt:=txt+cadena[i];
  cadena:=txt;
End;

Procedure CorrigeRespuesta;
Begin
  CompactaEspacios(respuesta);
  CompactaEspacios(solucion);
  If (respuesta=solucion) Then correcta(True)

```

```
    Else correcta(False);
End;

Procedure EnviaSolucion;
Var i:Integer;
Begin
    WriteLn(solucion);
End;

Begin

    { Obtiene los parámetros de entrada }
    programa:=ParamStr(0);
    accion:=ParamStr(1);
    Val(ParamStr(2),semilla,error);
    If (accion='/c') Or (accion='/C') Then
        For i:=3 To ParamCount Do Begin
            respuesta:=respuesta+ParamStr(i);
            If (i<ParamCount) Then respuesta:=respuesta+' ';
        End;

    { Genera los datos aleatorios }
    GeneraDatos;

    { En caso de acción '/e', genera el enunciado }
    If (accion='/e') Or (accion='/E') Then GeneraHTML;

    { En caso de acción '/c', corrige la respuesta }
    If (accion='/c') Or (accion='/C') Then CorrigeRespuesta;

    { En caso de acción '/s', devuelve la solución }
    If (accion='/s') Or (accion='/S') Then EnviaSolucion;

End.
```

8. Pregunta de selección simple (ejemplo 2)

En este ejemplo se genera una pregunta de **selección simple** en la que tanto una parte del enunciado (el nombre de un compuesto químico) como las posibles opciones de respuesta son variables. En este caso se juega con ocho pares de nombre de elementos químicos (ácidos con azufre) y las formulas químicas correspondientes. En la Figura 13 se muestran dos posibles variantes de la pregunta.

Enunciado
La formula quimica del acido sulfuroso es:

Respuesta
Selecciona la respuesta correcta:

1. ☐ H_2SO_2
2. ☐ H_2SO_3
3. ☐ $\text{H}_2\text{S}_2\text{O}_6$
4. ☐ $\text{H}_2\text{S}_2\text{O}_7$
5. ☐ $\text{H}_2\text{S}_2\text{O}_5$
6. ☐ Ninguna de las anteriores

Enunciado
La formula quimica del acido disulfuroso o pirodisulfuroso es:

Respuesta
Selecciona la respuesta correcta:

1. ☐ $\text{H}_2\text{S}_2\text{O}_3$
2. ☐ $\text{H}_2\text{S}_2\text{O}_7$
3. ☐ $\text{H}_2\text{S}_2\text{O}_6$
4. ☐ H_2SO_2
5. ☐ $\text{H}_2\text{S}_2\text{O}_4$
6. ☐ Ninguna de las anteriores

Figura 13. Ejemplo de pregunta de enunciado variable de selección simple

A continuación se muestra el código fuente del correspondiente programa generador/corrector. Como en los ejemplos anteriores el tipo de letra negrita indica el código a modificar del programa para generar otra pregunta similar

```

{ *****
  PREGUNTA:      quimica2.pas
  AUTOR:   agarcia@etsii.upm.es
  REVISION: 22.11.2002
  DESCRIPCION: Nombres de oxoacidos de azufre y formulas quimicas
  PARAMETROS VARIANTES: nombres - formulas
  ***** }

Program
  quimica2;

Const
  NOpciones=6;      { N$ de opciones que aparecen en el enunciado }
  NDefiniciones=8;  { N$ de pares formulas-nombre }

  secciones:Array[0..NDefiniciones-1] Of String=
    ('sulfuroso',
     'disulfuroso o pirodisulfuroso',
     'sulfurico',
     'pirosulfurico o disulfurico',
     'sulfoxilico',
     'ditionoso',
     'ditionico',
     'tiosulfurico');
  definiciones:Array[0..NDefiniciones-1] Of String=
    ('H<sub>2</sub>SO<sub>3</sub>',
     'H<sub>2</sub>S<sub>2</sub>O<sub>5</sub>',
     'H<sub>2</sub>SO<sub>4</sub>',
     'H<sub>2</sub>S<sub>2</sub>O<sub>7</sub>',
     'H<sub>2</sub>SO<sub>2</sub>',
     'H<sub>2</sub>S<sub>2</sub>O<sub>4</sub>',
     'H<sub>2</sub>S<sub>2</sub>O<sub>6</sub>',
     'H<sub>2</sub>S<sub>2</sub>O<sub>3</sub>');

Var
  i,j,error:Integer;
  semilla:LongInt;
  programa,accion,respuesta,enunciado,entrada,solucion:String;
  { Datos especificos }
  sol:Integer;

Procedure GeneraDatos;
Var flag:Boolean;
  txt:String;
Begin
  RandSeed:=semilla;
  { generaci n los datos aleatorios del enunciado }
  sol:=1+Random(NOpciones);
  Str(sol,solucion);
  For i:=0 To NDefiniciones-1 Do Begin
    j:=Random(NDefiniciones-1);
    txt:=secciones[i];
    secciones[i]:=secciones[j];
    secciones[j]:=txt;
    txt:=definiciones[i];
    definiciones[i]:=definiciones[j];
    definiciones[j]:=txt;
  End;
End;

Procedure GeneraEnunciado;

```

```

Begin
  Write('La formula quimica del acido ');
  Write(secciones[sol]);
  Write(' es:');
End;

Procedure GeneraInput;
Begin
  WriteLn('Selecciona la respuesta correcta:<br>');
  WriteLn('<table border=0 cellpadding=0 cellspacing=0>');
  For i:=1 To NOpciones-1 Do Begin
    Write('<tr><td id=t1>');
    Write(i, '. <input id=t1 type=radio name=respuesta value=', i, '> ', defini
ciones[i]);
    WriteLn('</td></tr>');
  End;
  Write('<tr><td id=t1>');
  Write(NOpciones, '. <input id=t1 type=radio name=respuesta value=', NOpciones, '>
');
  Write(' Ninguna de las anteriores');
  WriteLn('</td></tr>');
  WriteLn('</table>');
End;

Procedure GeneraHTML;
Begin
  WriteLn('<center><table border=0 cellpadding=0 cellspacing=8
width="100%"><tr>');
  WriteLn('<td id="t1" valign="top">');
  WriteLn('<b>Enunciado</b><br>');
  WriteLn('<div id="t1" style="text-align: justify;">');
  GeneraEnunciado;
  WriteLn('</div><br></td></tr><tr><td id="t1" valign="top">');
  WriteLn('<b>Respuesta</b><br>');
  GeneraInput;
  WriteLn('</td></tr></table></center>');
End;

Procedure Correcta(flag:Boolean);
Begin
  If flag Then WriteLn('OK. Respuesta correcta')
  Else WriteLn('ERROR. Respuesta incorrecta');
End;

Procedure CompactaEspacios(Var cadena:String);
Var i:Integer;
    txt:String;
Begin
  txt:='';
  For i:=1 To Length(cadena) Do
    If (cadena[i]<>' ') Then txt:=txt+cadena[i];
  cadena:=txt;
End;

Procedure CorrigeRespuesta;
Begin
  CompactaEspacios(respuesta);
  CompactaEspacios(solucion);
  If (respuesta=solucion) Then correcta(True)
  Else correcta(False);
End;

```



```
Procedure EnviaSolucion;
Var i:Integer;
Begin
  WriteLn(solucion);
End;

Begin

  { Obtiene los parámetros de entrada }
  programa:=ParamStr(0);
  accion:=ParamStr(1);
  Val(ParamStr(2),semilla,error);
  If (accion='/c') Or (accion='/C') Then
    For i:=3 To ParamCount Do Begin
      respuesta:=respuesta+ParamStr(i);
      If (i<ParamCount) Then respuesta:=respuesta+' ';
    End;

  { Genera los datos aleatorios }
  GeneraDatos;

  { En caso de acción '/e', genera el enunciado }
  If (accion='/e') Or (accion='/E') Then GeneraHTML;

  { En caso de acción '/c', corrige la respuesta }
  If (accion='/c') Or (accion='/C') Then CorrigeRespuesta;

  { En caso de acción '/s', devuelve la solución }
  If (accion='/s') Or (accion='/S') Then EnviaSolucion;

End.
```

9. Pregunta de selección múltiple (ejemplo 3)

En este ejemplo se genera una pregunta de **selección múltiple** en la que las posibles opciones de respuesta son variables. En este caso se juega con seis posibles respuestas correctas (símbolos químicos correspondientes a gases nobles) y otras seis que no lo son. En la Figura 14 se muestran dos posibles variantes de la pregunta.

Enunciado
Indicar cuales de los siguientes simbolos corresponden a **gases nobles**:

Respuesta
Selecciona las respuestas correctas:

1. ☐ **F**
2. ☐ **A**
3. ☐ **N**
4. ☐ **Be**
5. ☐ Ninguno de los anteriores

Enunciado
Indicar cuales de los siguientes simbolos corresponden a **gases nobles**:

Respuesta
Selecciona las respuestas correctas:

1. ☐ **Ne**
2. ☐ **Kr**
3. ☐ **Rn**
4. ☐ **Xe**
5. ☐ Ninguno de los anteriores

Figura 14. Ejemplo de pregunta de enunciado variable de selección múltiple

A continuación se muestra el código fuente del correspondiente programa generador/corrector. Como en los ejemplos anteriores el tipo de letra negrita indica el código a modificar del programa para generar otra pregunta similar

```

{ *****
  PREGUNTA: quimica3.pas
  AUTOR: agarcia@etsii.upm.es
  REVISION: 22.11.2002
  DESCRIPCION: Simbolos correspondientes a  de gases nobles
  ENUNCIADO: Indicar cuales de los siguientes simbolos corresponden
a gases nobles
  ***** }

Program
  quimica3;

Const
  NOpciones=5; { Numero de opciones que aparecen en el enunciado }
  NCorrectas=6; { Total de posibles respuestas correctas }
                { = Total de posibles respuestas incorrectas }
  correctas:Array[0..NCorrectas-1] Of String=
    ('He','Ne','A','Kr','Xe','Rn');
  incorrectas:Array[0..NCorrectas-1] Of String=
    ('H','F','Li','Be','O','N');

Type
  tOpcion=Record
    texto:String;
    correcta:Boolean;
  End;

Var
  start,pointer:Integer;
  i,j,error:Integer;
  semilla:LongInt;
  programa,accion,respuesta,enunciado,entrada,solucion:String;
  { Datos especificos }
  opciones:Array[0..NOpciones] Of tOpcion;

Procedure GeneraDatos;
Var prob:Integer;
  flag:Boolean;
  txt:String;
Begin
  RandSeed:=semilla;
  { generacin los datos aleatorios del enunciado }
  {desordena las listas de correctas e incorrectas}
  For i:=1 To NCorrectas-1 Do Begin
    j:=Random(NCorrectas-1);
    txt:=correctas[i];
    correctas[i]:=correctas[j];
    correctas[j]:=txt;
    j:=Random(NCorrectas-1);
    txt:=incorrectas[i];
    incorrectas[i]:=incorrectas[j];
    incorrectas[j]:=txt;
  End;
  solucion:='';
  flag:=True;
  { proporcion de correctas }
  prob:=Random(100);
  For i:=1 To NOpciones-1 Do Begin
    If (Random(100)<prob) Then Begin
      If Not(flag) Then solucion:=solucion+', ';
      Str(i,txt);

```

```

        solucion:=solucion+txt;
        flag:=False;
        opciones[i].texto:='<code>'+correctas[i]+'</code>';
        opciones[i].correcta:=True;
    End
Else Begin
    opciones[i].texto:='<code>'+incorrectas[i]+'</code>';
    opciones[i].correcta:=False;
End;
End;
opciones[NOPciones].texto:='Ninguno de los anteriores';
opciones[NOPciones].correcta:=flag;
If flag Then Begin
    Str(NOPciones,txt);
    solucion:=txt;
End;
End;

Procedure GeneraEnunciado;
Begin
    Write('Indicar cuales de los siguientes simbolos corresponden a ');
    WriteLn('<b>gases nobles</b>: ');
End;

Procedure GeneraInput;
Begin
    WriteLn('Selecciona las respuestas correctas:<br>');
    WriteLn('<table border=0 cellpadding=0 cellspacing=0>');
    For i:=1 To NOPciones Do Begin
        Write('<tr><td id=t1>');
        Write(i, '. <input id=t1 type=checkbox name=respuesta value=', i, '> ', opciones[i].texto);
        WriteLn('</td></tr>');
    End;
    WriteLn('</table>');
End;

Procedure GeneraHTML;
Begin
    WriteLn('<center><table border=0 cellpadding=0 cellspacing=8 width="100%"><tr>');
    WriteLn('<td id="t1" valign="top">');
    WriteLn('<b>Enunciado</b><br>');
    WriteLn('<div id="t1" style="text-align: justify;">');
    GeneraEnunciado;
    WriteLn('</div><br></td></tr><tr><td id="t1" valign="top">');
    WriteLn('<b>Respuesta</b><br>');
    GeneraInput;
    WriteLn('</td></tr></table></center>');
End;

Procedure Correcta(flag:Boolean);
Begin
    If flag Then WriteLn('OK. Respuesta correcta')
    Else WriteLn('ERROR. Respuesta incorrecta');
End;

Procedure CompactaEspacios(Var cadena:String);
Var i:Integer;
    txt:String;
Begin

```

```
txt:='';
For i:=1 To Length(cadena) Do
  If (cadena[i]<>' ') Then txt:=txt+cadena[i];
cadena:=txt;
End;

Procedure CorrigeRespuesta;
Begin
  CompactaEspacios(respuesta);
  CompactaEspacios(solucion);
  If (respuesta=solucion) Then correcta(True)
  Else correcta(False);
End;

Procedure EnviaSolucion;
Var i:Integer;
Begin
  WriteLn(solucion);
End;

Begin
  { Obtiene los parámetros de entrada }
  programa:=ParamStr(0);
  accion:=ParamStr(1);
  Val(ParamStr(2),semilla,error);
  If (accion='/c') Or (accion='/C') Then
    For i:=3 To ParamCount Do Begin
      respuesta:=respuesta+ParamStr(i);
      If (i<ParamCount) Then respuesta:=respuesta+' ';
    End;
  { Genera los datos aleatorios }
  GeneraDatos;
  { En caso de acción '/e', genera el enunciado }
  If (accion='/e') Or (accion='/E') Then GeneraHTML;
  { En caso de acción '/c', corrige la respuesta }
  If (accion='/c') Or (accion='/C') Then CorrigeRespuesta;
  { En caso de acción '/s', devuelve la solución }
  If (accion='/s') Or (accion='/S') Then EnviaSolucion;
End.
```

10. Pregunta de respuesta numérica entera (ejemplo 4)

En este ejemplo se genera una pregunta de respuestas **numérica entera** en la que parte del enunciado es variable: el número de átomos de carbono de un alcano. En la Figura 15 se muestran dos posibles variantes de la pregunta.

<p>Enunciado</p> <p>La molecula de un hidrocarburo lineal saturado o <i>alcano</i> tiene 11 atomos de carbono. Indicar el numero de atomos de hidrogeno que tiene la molecula</p> <p>Respuesta</p> <p>Introduce la respuesta: <input type="text"/></p>
<p>Enunciado</p> <p>La molecula de un hidrocarburo lineal saturado o <i>alcano</i> tiene 13 atomos de carbono. Indicar el numero de atomos de hidrogeno que tiene la molecula</p> <p>Respuesta</p> <p>Introduce la respuesta: <input type="text"/></p>

Figura 15. Ejemplo de pregunta de respuesta numérica entera

A continuación se muestra el código fuente del correspondiente programa generador/corrector. Como en los ejemplos anteriores el tipo de letra negrita indica el código a modificar del programa para generar otra pregunta similar

```

{ *****
  PREGUNTA: quimica4
  AUTOR: agarcia@etsii.upm.es
  REVISION: 22.11.2002
  DESCRIPCION: Numero de atomos de hidrogeno en hidrocarburo saturado
  PARAMETROS VARIANTES: Numero de atomos de carbono
  ***** }

Program
  quimica4;

Var
  i,error:integer;
  semilla:longint;
  solucion,programa,accion,respuesta,enunciado,entrada:string;
  respuestaCorrecta:integer;
  { Datos especificos }
  nAtomosC:byte;

Procedure GeneraDatos;
begin
  RandSeed:=semilla;
  { Generacion los datos aleatorios del enunciado }
  nAtomosC:=random(18)+2;
end;

Procedure GeneraEnunciado;
begin
  write('La molecula de un hidrocarburo lineal saturado o <i>alcano</i> ');
  write('tiene ',nAtomosC,' atomos de carbono. Indicar el <b>numero</b> ');
  writeln('de atomos de hidrogeno que tiene la molecula');
end;

Procedure GeneraInput;
begin
  writeln('Introduce la respuesta: <input id="t1" type="text" name="respuesta"
size="
16" maxlength="32">');
end;

Procedure GeneraHTML;
begin
  writeln('<center><table border=0 cellpadding=0 cellspacing=8
width="100%"><tr>');
  writeln('<td id="t1" valign="top">');
  writeln('<b>Enunciado</b><br>');
  writeln('<div id="t1" style="text-align: justify;">');
  GeneraEnunciado;
  writeln('</div><br></td></tr><tr><td id="t1" valign="top">');
  writeln('<b>Respuesta</b><br>');
  GeneraInput;
  writeln('</td></tr></table></center>');
end;

Procedure Correcta(flag:boolean);
begin
  if flag then writeln('OK. Respuesta correcta')
  else writeln('ERROR. Respuesta incorrecta');
end;

Procedure CorrigeRespuesta;

```

```

var s:string;
begin
  str(respuestaCorrecta,s);
  if (respuesta=s) then correcta(true)
  else correcta(false);
end;

Procedure EnviaSolucion;
Begin
  WriteLn(solucion);
End;

BEGIN

  { Obtiene los par metros de entrada }
  programa:=ParamStr(0);
  accion:=ParamStr(1);
  val(ParamStr(2),semilla,error);
  if (accion='/c') or (accion='/C') then respuesta:=ParamStr(3);

  { Genera los datos aleatorios }
  GeneraDatos;

  { Halla la solucion correcta }
respuestaCorrecta:=nAtomosC*2+2;

  { En caso de acciøn '/e', genera el enunciado }
  if (accion='/e') or (accion='/E') then GeneraHTML;

  { En caso de acciøn '/c', corrige la respuesta }
  if (accion='/c') or (accion='/C') then CorrigeRespuesta;

  str(respuestaCorrecta,solucion);

  { En caso de acciøn '/s', devuelve la solución }
  If (accion='/s') Or (accion='/S') Then EnviaSolucion;

END.

```


11. Pregunta de respuesta numérica real (ejemplo 5)

En este ejemplo se genera una pregunta de respuestas **numérica real** en la que parte del enunciado es variable: la cantidad de óxido mercuríco que participa en una reacción. En la Figura 16 se muestran dos posibles variantes de la pregunta.

<p>Enunciado</p> <p>La reaccion correspondiente a la descomposicion del óxido mercurico es:</p> $2 \text{HgO} = 2 \text{Hg} + \text{O}_2$ <p>Hallar la cantidad de oxigeno (en gramos) que puede obtenerse al calentar 37 g de HgO</p> <p>Nota: la precision del valor pedido es ± 0.01</p> <p>Respuesta</p> <p>Introduce la respuesta: <input type="text"/></p>
<p>Enunciado</p> <p>La reaccion correspondiente a la descomposicion del óxido mercurico es:</p> $2 \text{HgO} = 2 \text{Hg} + \text{O}_2$ <p>Hallar la cantidad de oxigeno (en gramos) que puede obtenerse al calentar 70 g de HgO</p> <p>Nota: la precision del valor pedido es ± 0.01</p> <p>Respuesta</p> <p>Introduce la respuesta: <input type="text"/></p>

Figura 16. Ejemplo de pregunta de respuesta numérica real

A continuación se muestra el código fuente del correspondiente programa generador/corrector. Como en los ejemplos anteriores el tipo de letra negrita indica el código a modificar del programa para generar otra pregunta similar

```

{ *****
  PREGUNTA: quimica5.pas
  AUTOR: agarcia@etsii.upm.es
  REVISION: 22.11.2002
  DESCRIPCION: Resultado de la reaccion de descomposicion del HgO
  PARAMETROS VARIANTES: Cantidad inicial de HgO
  ***** }

Program
  quimica5;

Const
  precision = 0.01;

Var
  i,error:integer;
  semilla:longint;
  solucion,programa,accion,respuesta,enunciado,entrada:string;
  respuesta_Correcta:real;
  { Variables para datos especificos de esta pregunta}
  cantidad_inicial:integer;

Procedure GeneraDatos;
begin
  RandSeed:=semilla;
  { Generacion los datos aleatorios del enunciado }
  cantidad_inicial:=random(81)+20; { Para que sean entre 20 y 100 gramos }
end;

Procedure GeneraEnunciado;
begin
  write('La reaccion correspondiente a la descomposicion del <b>oxido ');
  write('mercurico</b> es:<br><br>');
  write('<center>2 HgO = 2 Hg + O<sub>2</sub></center><br>');
  write('Hallar la <b>cantidad</b> de oxigeno (en gramos) que puede obtenerse');
  writeln(' al calentar ',cantidad_inicial,' g de HgO<br>');
  writeln('Nota: la precision del valor pedido es ±0.01');
end;

Procedure GeneraInput;
begin
  writeln('Introduce la respuesta: <input id="t1" type="text" name="respuesta"
size="
16" maxlength="32">');
end;

Procedure GeneraHTML;
begin
  writeln('<center><table border=0 cellpadding=0 cellspacing=8
width="100%"><tr>');
  writeln('<td id="t1" valign="top">');
  writeln('<b>Enunciado</b><br>');
  writeln('<div id="t1" style="text-align: justify;">');
  GeneraEnunciado;
  writeln('</div><br></td></tr><tr><td id="t1" valign="top">');
  writeln('<b>Respuesta</b><br>');
  GeneraInput;
  writeln('</td></tr></table></center>');
end;

Procedure Correcta(flag:boolean);

```

```
begin
  if flag then writeln('OK. Respuesta correcta')
  else writeln('ERROR. Respuesta incorrecta');
end;

Procedure CorrigeRespuesta;
var r:real; code:integer;
begin
  { Conversion de la respuesta de tipo cadena (string) a un valor r real: }
  val(respuesta,r,code);
  { Si code vale cero entonces no ha habido problema en la conversion }
  if (code=0) and (abs(respuesta_Correcta-r)<precision)
    then correcta(true)
    else correcta(false);
end;

Procedure EnviaSolucion;
Begin
  WriteLn(solucion);
End;

BEGIN

  { Obtiene los par metros de entrada }
  programa:=ParamStr(0);
  accion:=ParamStr(1);
  val(ParamStr(2),semilla,error);
  if (accion='/c') or (accion='/C') then respuesta:=ParamStr(3);

  { Genera los datos aleatorios }
  GeneraDatos;

  { Halla la solucion correcta }
  { Peso molecular del oxigeno: 32 }
  { Peso molecular del HgO: 433.32 }
  respuesta_Correcta:=cantidad_inicial*32/433.32;

  { En caso de acciøn '/e', genera el enunciado }
  if (accion='/e') or (accion='/E') then GeneraHTML;

  { En caso de acciøn '/c', corrige la respuesta }
  if (accion='/c') or (accion='/C') then CorrigeRespuesta;

  str(respuesta_Correcta,solucion);

  { En caso de acciøn '/s', devuelve la solución }
  If (accion='/s') Or (accion='/S') Then EnviaSolucion;

END.
```

12. Conclusión

En este manual se ha mostrado el modo de funcionamiento y de generación de preguntas de enunciado variable para la plataforma AulaWeb. Se facilitan varios ejemplos de programas que permiten la implementación de preguntas de este tipo con diferentes clases de respuestas.

La principal ventaja de este tipo de preguntas radica en la posibilidad de generar un número prácticamente infinito de versiones de un mismo problema lo que reduce la posibilidad de copia entre alumnos y de *agotamiento* de la base de datos del sistema. El mayor inconveniente es la complejidad para su implementación.

Bibliografía

Manuales de utilización

1. A. García-Beltrán, R. Martínez. *Manual del Administrador General - AulaWeb 2004*, 110 pp, febrero de 2004. ISBN: 84-688-5871-4
2. R. Martínez, A. García-Beltrán. *AulaWeb: Manual del profesor*, 94 pp, diciembre de 2002. ISBN: 84-7484-154-2
3. A. García-Beltrán, R. Martínez. *AulaWeb: Manual del alumno*, 100 pp, junio de 2002. ISBN: 84-7484-148-8
4. R. Martínez, A. García-Beltrán, A. Alonso y J. A. Criado. *AulaWeb - Student Guide*, 23 pp, febrero de 2005. ISBN: 84-689-8653-3
5. A. García-Beltrán y R. Martínez. *AulaWeb - Publicación de contenidos en formato SCORM*, 54 pp, junio de 2007. ISBN: 978-84-690-8575-2
6. R. Martínez y A. García-Beltrán. *AulaWeb - Módulo de trabajo colaborativo*, 20 pp, febrero de 2008. ISBN: 978-84-691-2451-2

Informes de uso

7. A. García-Beltrán, R. Martínez, Utilización de AulaWeb en la asignatura de Informática durante el primer cuatrimestre del curso 2000-01, 44 pp, marzo de 2001. ISBN: 84-688-5361-5
8. A. García-Beltrán, R. Martínez. Utilización de AulaWeb como sistema de e-learning en la asignatura de Informática durante el primer cuatrimestre del curso 2001-02, febrero de 2002. 68 pp, ISBN: 84-688-5362-3
9. A. García-Beltrán, R. Martínez. Utilización de AulaWeb como sistema de e-learning en la asignatura de Informática durante el primer cuatrimestre del curso 2002-03, 58 pp, febrero de 2003. ISBN: 84-688-5363-1
10. A. García-Beltrán, R. Martínez. Utilización de AulaWeb como sistema de b-learning en la asignatura de Informática durante el primer cuatrimestre del curso 2003-04, 79 pp, marzo de 2004. ISBN: 84-688-6279-7
11. A. García-Beltrán, R. Martínez. Utilización de AulaWeb como sistema de b-learning en el curso 2004-05, 85 pp, marzo de 2005. ISBN: 84-689-1130-5
12. A. García-Beltrán. Innovación Educativa en la Docencia Complementaria de Informática en el curso 2004-05, 36 pp, junio de 2005. ISBN: 978-84-690-8576-9
13. J. M. Arranz, A. García-Beltrán, J. A. Jaén, R. Martínez y S. Tapia. Utilización de AulaWeb como apoyo docente en las asignaturas de Informática en el curso 2005-06, 76 pp, marzo de 2006. ISBN: 84-689-8163-X
14. A. García-Beltrán, R. Martínez, J.A. Jaén, S. Tapia y J.M. Arranz. Utilización de AulaWeb como apoyo docente en las asignaturas de Informática en el curso 2006-7, 81 pp, marzo de 2007. ISBN: 978-84-690-5191-7

Artículos en revistas

15. R. Martínez, A. García-Beltrán, J.A. Jaén, Un sistema WWW de ayuda a la formación para alumnos y profesores, Ingeniería I+D, 44, 327-329 (2000)
16. A. García-Beltrán y R. Martínez. Experiencia docente con AulaWeb en la asignatura de Informática, RED - Revista de Educación a Distancia, 11, 2004.
17. A. García-Beltrán, R. Martínez, J.A. Criado y A. Alonso. Taxonomía de los contenidos formativos en formato electrónico empleados en las asignaturas impartidas en la ETSII-UPM, RED - Revista de Educación a Distancia, número monográfico III, (2005, Febrero), ISSN: 1578-7680
18. A. García-Beltrán, R. Martínez, J.A. Jaén y S. Tapia. La autoevaluación como actividad docente en entornos virtuales de aprendizaje/enseñanza, RED - Revista de Educación a Distancia, número monográfico VI, (2006, Septiembre), ISSN: 1578-7680
19. A. García-Beltrán, R. Martínez, J.A. Jaén y S. Tapia. La autoevaluación como actividad docente en entornos virtuales de aprendizaje/enseñanza, RED - Revista de Educación a Distancia, número monográfico VI, (2006, Septiembre), ISSN: 1578-7680

20. A. García-Beltrán y R. Martínez. Web assisted assessment in computer programming learning using AulaWeb, *International Journal of Engineering Education*, 22-5 (2006, September), ISSN: 0949-149X
21. A. García-Beltrán, S. Tapia, R. Martínez y M. González. Automatic assessment of multi-language programming questions using AulaWeb, *WSEAS Transactions on Computers*, Issue 3, Vol. 6 449-454 (2007), ISSN: 1109-2750

Ponencias y comunicaciones en congresos nacionales

22. M. Boyer. *Las Bibliotecas de la Universidad Politécnica de Madrid en AulaWeb*, Actas de la Jornada de Nuevas Tecnologías en la Innovación Educativa, 218-221 (2003), Madrid (España), 25 de septiembre de 2003, ISBN: 84-7484-157-7
23. J.A Sánchez, J.L. Bravo, M. Farjas, J. Prieto. *Innovaciones didácticas en AulaWeb: el estudio de casos en la titulación de Ingeniería Técnica Topográfica*, Actas de la Jornada de Nuevas Tecnologías en la Innovación Educativa, 209-217 (2003), Madrid (España), 25 de septiembre de 2003, ISBN: 84-7484-157-7
24. J. M. Pérez García. *Experiencia del uso de AulaWeb en las asignaturas de Tecnología Mecánica I*, Actas de la Jornada de Nuevas Tecnologías en la Innovación Educativa, 161-168 (2003), Madrid (España), 25 de septiembre de 2003, ISBN: 84-7484-157-7
25. J.L. Bravo, J.A Sánchez, M. Farjas, J. Prieto. *Aplicación experimental de la plataforma AulaWeb en Topografía*, Actas de la Jornada de Nuevas Tecnologías en la Innovación Educativa, 143-152 (2003), Madrid (España), 25 de septiembre de 2003, ISBN: 84-7484-157-7
26. F. M. Sánchez Moreno, A. García Dopico. *Experiencia docente con AulaWeb en la asignatura de sistemas Operativos* en el curso 2002-03, Actas de la Jornada de Nuevas Tecnologías en la Innovación Educativa, 118-127 (2003), Madrid (España), 25 de septiembre de 2003, ISBN: 84-7484-157-7
27. A. García-Beltrán, R. Martínez. *Uso de AulaWeb en la asignatura de Informática en la ETSII-UPM en el curso 2002-03*, Actas de la Jornada de Nuevas Tecnologías en la Innovación Educativa, 153-160 (2003), Madrid (España), 25 de septiembre de 2003, ISBN: 84-7484-157-7
28. R. Martínez, A. García-Beltrán. *Experiencia en el uso de las Nuevas Tecnologías en la UPM*, Actas de la Jornada de Nuevas Tecnologías en la Innovación Educativa, 41-48 (2003), Madrid (España), 25 de septiembre de 2003, ISBN: 84-7484-157-7
29. R. Martínez, A. García-Beltrán. *El Sistema AulaWeb para la Empresa y la Universidad*, Actas del SIT'2002 - Simposio en Informática y Telecomunicaciones, 271-280 (2002)
30. R. Martínez, A. García-Beltrán. *La plataforma de Tele-Educación AulaWeb*, SIT '02 - Simposio de Informática y Telecomunicaciones, Sevilla (España), septiembre 25-27, 2002.
31. A. García-Beltrán, R. Martínez. *Experiencias del uso de AulaWeb en la Universidad Politécnica de Madrid*, II Congreso de Aplicación de las Nuevas Tecnologías en la Docencia Presencial y E-Learning, Universidad Cardenal Herrera-CEU, Valencia (España), septiembre 10-11, 2002.
32. A. García-Beltrán, R. Martínez. *Enseñar/Aprender Programación Empleando Aulaweb*, Simposio en Informática y Telecomunicaciones, SIT 2001, La Coruña (España), 12-14 de septiembre de 2001.
33. J.L. Bravo, J.A. Sánchez y M. Farjas. El uso de sistemas de b-learning en la enseñanza universitaria, Coloquio Aulas con Software 2004, SIMO (Madrid), 12 de Noviembre 2004.
34. R. Martínez, A. García-Beltrán. *AulaWeb: Una plataforma e/b-Learning para la UPM*, VI Congreso de Tecnologías Aplicadas a la Enseñanza de la Electrónica, TAEE 2004, Valencia (España), 14-16 de julio, 2004.
35. A. García-Beltrán, R. Martínez. *Taxonomía de los contenidos formativos en formato electrónico empleados en las asignaturas impartidas en la ETSII-UPM*, I Simposio Pluridisciplinar sobre Diseño, Evaluación y Descripción de Contenidos Educativos Reutilizables, SPDECE 2004, Guadalajara (España), octubre 20-22, 2004.
36. A. García-Beltrán, R. Martínez. *Utilización de AulaWeb como apoyo docente para aprender a programar*, I Simposio Nacional de Tecnologías de la Información y de las Comunicaciones en la Educación, SINTICE 2005 (I Congreso Español de Informática, CEDI 2005), Granada (España), 13-16 de septiembre, 2005. Publicado en Actas del Simposio, pág. 37-44, ISBN: 84-9732-437-4
37. J. A. Jaén, R. Martínez, A. García-Beltrán, S. Tapia, J.M. Arranz. *Uso de AulaWeb como herramienta de b-Learning en la Asignatura de Informática en la ETSII-UPM*, I Jornadas de Innovación Educativa de la Escuela Politécnica Superior de Zamora, Zamora (España), 20-22 de junio, 2006. Publicado en Actas de las Jornadas, pág. 458-472, ISBN: 84-689-9304-2
38. M. J. Villamide, M. R. Alvir, Y. Alegre, J. García, N. Nicodemus. *El uso del sistema b-Learning mediante la plataforma AulaWeb en la asignatura troncal Zootécnica I en la E.T.S. de Ingenieros Agrónomos de Madrid*, I Jornadas de Innovación Educativa de la Escuela Politécnica Superior de Zamora, Zamora (España), 20-22 de junio, 2006. Publicado en Actas de las Jornadas, pág. 480-487, ISBN: 84-689-9304-2

39. A. García-Beltrán, J. A. Jaén, R. Martínez, S. Tapia. Programación Orientada a Objetos para Alumnos de Ingeniería en Automática y Electrónica Industrial, I VII Congreso TAEF de la UPM, Madrid (España), 20-22 de julio, 2006. Publicado en Resúmenes de los Trabajos del VII Congreso TAEF, pág. 111-112, ISBN: 84-689-9590-8
40. M. González, A. García-Beltrán, R. Martínez, S. Tapia. Preguntas autoevaluables de programación en C/C++ y Java en el sistema de eLearning AulaWeb, I VII Congreso TAEF de la UPM, Madrid (España), 20-22 de julio, 2006. Publicado en Resúmenes de los Trabajos del VII Congreso TAEF, pág. 141-142, ISBN: 84-689-9590-8
41. A. García-Beltrán y R. Martínez. ¿Están ya los alumnos cansados de las nuevas tecnologías para la innovación educativa como apoyo en el aprendizaje?, I Jornada Internacional UPM sobre Innovación Educativa y Convergencia Europea 2007 (inece'97), Madrid, 11-13 de diciembre de 2007. Publicado en Resúmenes de las Comunicaciones de la Jornada
42. J. Oñoro, J. R. Ibars, M. Roso y A. Portolés. Análisis de la mejora del rendimiento académico con la implantación de nuevos recursos metodológicos a través de la plataforma AulaWeb, I Jornada Internacional UPM sobre Innovación Educativa y Convergencia Europea 2007 (inece'97), Madrid, 11-13 de diciembre de 2007. Publicado en Resúmenes de las Comunicaciones de la Jornada
43. J. Rodríguez Martín y M. C. Huerta Gómez de Merodio. Herramientas didácticas para el estudio personal mediante el uso intensivo de AulaWeb, I Jornada Internacional UPM sobre Innovación Educativa y Convergencia Europea 2007 (inece'97), Madrid, 11-13 de diciembre de 2007. Publicado en Resúmenes de las Comunicaciones de la Jornada
44. M. J. Villamide, R. Carabaño, J. García, M. Alvir, J. González, N. Nicodemus, I. Solís, A. Saiz y D. Menoyo. Las plataformas educativas y la evaluación continua en asignaturas troncales: hacia la convergencia europea, I Jornada Internacional UPM sobre Innovación Educativa y Convergencia Europea 2007 (inece'97), Madrid, 11-13 de diciembre de 2007. Publicado en Resúmenes de las Comunicaciones de la Jornada
45. D. J. Muñoz, A. García-Beltrán, R. Martínez y J. M. Muñoz-Guijosa. Implementación de un Módulo de Gestión de Contenidos SCORM en la Plataforma AulaWeb, IV Simposio Pluridisciplinar sobre Diseño, Evaluación y Desarrollo de Contenidos Educativos Reutilizables (SPDECE07), Bilbao, 19-21 de septiembre de 2007. Publicado en Proc. del Simposio en formato CD-ROM. ISBN: 978-84-8373-992-1
46. M. J. Gallego y V. Gámiz. La plataforma AulaWeb en la enseñanza práctica de los estudiantes de Educación, eUniversaLearning: Congreso Internacional de Tecnología, Formación y Comunicación (eUniversaLearning 07), Salamanca, 12-15 de septiembre de 2007. Publicado en Actas del 1er Congreso Internacional. Salamanca. ISBN: 978-84-930218-4-9
47. M. J. Gallego y V. Gámiz. Ayudando al estudiante en prácticas a través del empleo de las TIC: El caso de la plataforma AulaWeb, II Jornadas Internacionales sobre políticas educativas para la sociedad del conocimiento (Sitio Web), Palacio de Congresos de Granada, 7-9 de marzo de 2007.

Ponencias y comunicaciones en congresos internacionales

48. A. García-Beltrán y R. Martínez. Challenges of a blended e-learning system in traditional engineering faculties, Advances in Technology-Based Education: Toward A Knowledge-Based Society. Proceedings of 2nd International Conference on Multimedia and Information & Communication Technologies in Education, Badajoz, Spain, December 3-6th (2003), Vol. III, pp. 1960-1963. ISBN: 84-96212-12-2
49. A. López, J. C. Pérez, C. Zoido, P. Avendaño, T. Hernández, E. Villalar, J.A. Martín, J.A. Criado, R. Martínez y A. García-Beltrán. User Interface and features of AulaWeb e-learning System, Advances in Technology-Based Education: Toward A Knowledge-Based Society. Proceedings of 2nd International Conference on Multimedia and Information & Communication Technologies in Education, Badajoz, Spain, December 3-6th (2003), Vol. III, pp. 1855-1859. ISBN: 84-96212-12-2
50. R. Martínez, A. García-Beltrán. Integración de las TIC en la Universidad Politécnica de Madrid, Actas de la conferencia OnLine Educa Barcelona, 59-63 (2003). ISBN: 3-9808909-1-0
51. A. García-Beltrán, R. Martínez. Desarrollo e implantación de un sistema WWW de gestión del aprendizaje/enseñanza en un centro universitario, Actas del Congreso CIVE 2003, ISBN: 84-7632-822-2
52. A. García-Beltrán, R. Martínez. The Role of Self-Assessment in AulaWeb e-learning System, 2002 EDEN - European Distance Education Network- Annual Conference, Granada (España), junio 16-19, 2002.
53. A. García-Beltrán, R. Martínez. Utilización de AulaWeb como sistema de apoyo en la impartición de asignaturas de informática en la ETSII-UPM, Conferencia Internacional de Tecnología, Educación y Desarrollo Sostenible, EDUTEC 2001, Murcia (España) Septiembre 17-19, 2001.
54. R. Martínez, J.A. Jaén, J.A. Criado, B. Suárez, J.A. Martín, J.M. Arranz, A. García-Beltrán. Desarrollo de aplicaciones informáticas para la educación. Conferencia Internacional de Tecnología, Educación y Desarrollo Sostenible, EDUTEC 2001, Murcia (España) Septiembre 17-19, 2001.

55. R. Martínez, A. García-Beltrán. AulaWeb: a WWW-Based Course-Support System with Self-Assessment and Student Tracking, World Conference on Educational Multimedia, Hypermedia and Telecommunications, EDMEDIA 2001, Tampere, Finlandia, Junio 25-30, 2001.
56. R. Martínez, A. García-Beltrán. Líneas de Actuación y Metodología de Uso de la Plataforma de Tele-Educación AulaWeb en la Enseñanza Presencial Universitaria, IADIS Conferencia Ibero-Americana WWW/Internet 2004, Madrid (España), 7-8 de octubre, 2004.
57. A. García-Beltrán, R. Martínez. Spread of an e-Learning System in a Polytechnical University, IADIS International Conference, e-Society 2004, Ávila (Spain), julio 16-19, 2004.
58. R. Martínez, A. García-Beltrán, J.A. Criado, P. Contreras, M. Iglesias, A. Rodelgo. Auto-aprendizaje y Evaluación sobre la Plataforma AulaWeb a Través de Dispositivos Portátiles, 5ª Conferencia Internacional de la Educación y la Formación Basada en las Tecnologías, OnLine Educa Madrid 2005, Punto de encuentro entre Europa y Latinoamérica, Madrid (España), 11-13 de mayo de 2005. Publicado en Actas de OnLine Educa Madrid 2005, pág. 85-86, ISBN: 3-9808909-8-8
59. R. Martínez, A. García-Beltrán, J.A. Criado, P. Contreras, M. Iglesias, A. Rodelgo. Auto-aprendizaje y Evaluación sobre la Plataforma AulaWeb a Través de Dispositivos Portátiles, 5ª Conferencia Internacional de la Educación y la Formación Basada en las Tecnologías, OnLine Educa Madrid 2005, Punto de encuentro entre Europa y Latinoamérica, Madrid (España), 11-13 de mayo de 2005. Publicado en Actas de OnLine Educa Madrid 2005, pág. 85-86, ISBN: 3-9808909-8-8
60. A. García-Beltrán, R. Martínez, J.A. Jaén, S. Tapia, J.M. Arranz. Making good use of AulaWeb in Computer Science Learning-Teaching, ICECE 2005, International Conference on Engineering and Computer Education, 14-16 Noviembre 2005, Madrid (SPAIN), Publicado en Book of Abstracts, ISBN: 84-609-8149-5
61. R. Martínez, A. García-Beltrán, J.A. Criado, C. Moreno. The Statistical Module of AulaWeb: A Tool for improving e-Learning Quality, Informatics Education Europe, 9-10 Noviembre 2006, Montpellier (Francia).
62. A. García-Beltrán, R. Martínez, J.A. Jaén. B-Learning with Engineering Students in Computer Science Courses, Informatics Education Europe, 9-10 Noviembre 2006, Montpellier (Francia).
63. A. García-Beltrán, S. Tapia, R. Martínez, M. González. A Web-based Self-assessment System with Multi-language Programming Questions, 5th WSEAS International Conference on E-ACTIVITIES, Venecia (Italia), 20-22 de noviembre de 2006. Publicado en Proceedings of the WSEAS International Conference: 5th WSEAS Int. Conf. on E-ACTIVITIES, ISBN: 960-8457-51-3
64. S. Tapia, A. García-Beltrán, R. Martínez. Program Testing in a Self-Assessment System Through Internet, IADIS International Conference on Cognition and Exploratory Learning in Digital Age (CELDA 2006), Barcelona (España), 8-10 de diciembre de 2006. Publicado en Proc. of IADIS - CELDA 2006, pags. 419-422. ISBN: 972-8924-24-0
65. A. García-Beltrán, R. Martínez, J.A. Jaén. A Case of Study: Building and Implementing a b-Learning system in an Engineering Faculty, The 2nd International Conference on Engineering Education & Training, Kuwait (Kuwait), 9-11 de abril de 2007.